IJARSCT

ISSN (Online) 2581-9429

International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)

Impact Factor: 6.252

Volume 2, Issue 1, August 2022

Dynamic and Optimized Routing Scheme based on Fuzzy with Trustable Transmission in Wireless Sensor Networks

Christy Quintus T[1], Veerabagu P[2], Mathavi S[2], Daniel Das A[3]

PG Scholar, Department of Electronics and Communication Engineering[1]
Assistant Professor, Department of Electronics and Communication Engineering[2]
Assistant Professor, Department of Mechanical Engineering[3]
RVS College of Engineering, Dindigul, Tamilnadu, India[1,2]
Karpagam Academy of Higher Education, Coimbatore, Tamilnadu, India
Corresponding Author Email: quintus1812@gmail.com[1]

Abstract: *In this research work, an effective proposed scheme is used named, Dynamic and Optimized Routing Scheme with Secure Transmission in WSN to overcome above many in networks. It provides load balanced and optimized path selection using Secure Fuzzy Logic Optimization and finds a stable path between the source and destination meeting the delay requirement. To conquer security challenge, an active detection-based security and trust routing scheme named ActiveTrust is proposed for WSNs. The most important innovation of ActiveTrust is that it avoids attacks through the active creation of a number of detection routes to quickly detect and obtain nodal trust and thus improve the data route security. Our proposed scheme which maximizes end-to-end connectivity in the network and minimizes faults at link or/and node level. Secure Fuzzy Logic Optimization Routing (SFLOR) where the energy efficiency is taken as a major criterion for performing routing and deriving optimized path for data forwarding and processing to base node. The SFLOR generates a whole new path of routing by taking energy as fitness value to judge different path and choose best optimized path whose energy consumption is less as compared to other routing paths.*

Keywords: Wireless sensor networks, ActiveTrust, SFLOR, wireless video transmission, wireless multimedia sensor networks.

I. INTRODUCTION

A Wireless Sensor Network (WSN) consists of tiny nodes that are equipped with embedded computing devices interfaced with sensors or actuators. These sensor networks are a vital component of Internet of Things (IoT) and are characterized as constrained networks due to limited memory, computing and energy capability. Fuzzy logic is a form of many-valued logic in which the truth values of variables may be any real number between 0 and 1 inclusive. It is employed to handle the concept of partial truth, where the truth value may range between completely true and completely false. By contrast, in Boolean logic, the truth values of variables may only be the integer values 0 or 1.

It is based on the observation that people make decisions based on imprecise and non-numerical information, fuzzy models or sets are mathematical means of representing vagueness and imprecise information, hence the term fuzzy. These models have the capability of recognizing, representing, manipulating, interpreting, and utilizing data and information that are vague and lack certainty. A Wireless sensor network can be defined as a network of devices that can communicate the information gathered from a monitored field through wireless links. The data is forwarded through multiple nodes, and with a gateway, the data is connected to other networks like wireless Ethernet.

Currently, WSN (Wireless Sensor Network) is the most standard services employed in commercial and industrial applications, because of its technical development in a processor, communication, and low-power usage of embedded computing devices. The WSN is built with nodes that are used to observe the surroundings like temperature, humidity, pressure, position, vibration, sound etc. These nodes can be used in various real-time applications to perform various tasks like smart detecting, a discovery of neighbor node, data processing and storage, data collection, target tracking,

DOI: 10.48175/IJARSCT-5997

monitor and controlling, synchronization, node localization, and effective routing between the base station and nodes. Fig 1 shows the wireless sensor network architecture.

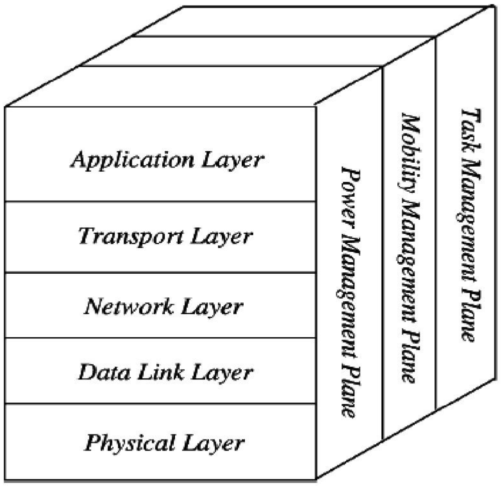

Fig. 1 Wireless Sensor Network Architecture

II. RESEARCH METHODOLOGIES

In this existing methodology, we used a novel framework for congestion detection name as Hybrid Congestion Control Mechanism (HCM) as dynamic optimized scheduling and congestion control for Wireless Multimedia Sensor Network (WMSN). Hybrid Congestion Control Mechanism includes techniques are Optimized Queue Scheduling, Hop-by-hop congestion control, and Congestion Aware Adaptive Routing. The existing algorithm is a hybrid congestion control protocol that considers not only the packets delivery rate but also retains the buffer size of each node. The proposed protocol may avoid packets drop due to traffic congestion and improve the network throughput.

2.1 Existing Methodology

In this proposed methodology, we used a novel framework for congestion detection name as Hybrid Congestion Control Mechanism (HCM) as dynamic optimized scheduling and congestion control for Wireless Multimedia Sensor Network (WMSN).Hybrid Congestion Control Mechanism includes techniques are Optimized Queue Scheduling, Hop-by-hop congestion control, and Congestion Aware Adaptive Routing. We investigate and measures the optimum performance of a congestion protocol during link and buffer congestion states; then establishes relationships between optimum transmission window and cache size at congestion states in WMSN; and provides a basis for future congestion control management that guarantees congestion avoidance while meeting the optimum detection and routing performance [11].

The proposed algorithm is a hybrid congestion control protocol that considers not only the packets delivery rate but also retains the buffer size of each node. The proposed protocol may avoid packets drop due to traffic congestion and improve the network throughput. Hybrid Congestion Control Mechanism (HCCM) – Dynamic Optimized Scheduling and Congestion control (DOSC).In many of the existing research works, delay performance is not taken into consideration. A general optimal solution can be got by means a duality approach that gives rise to the back-pressure algorithm and a congestion control element at the source node. Moreover, efforts have been focused on the development of less complexity and distributed scheduling algorithms, which can substitute the centralized back-pressure algorithm and still can accomplish decent and good throughput performance. Similar to the back-pressure algorithm, these low-complexity scheduling algorithms are generally based on the queue-length. However, the disadvantage of these approaches, is that the end-to-end delay arising from the resultant queue-length-based scheduling algorithm is very hard to be quantified, and there are proofs that, under few cases, there can be poor delay performance in back-pressure. In this research work, a window-based flow control algorithm and a virtual rate-based scheduling algorithm is proposed, which have huge differences with back-pressure. These variables rm are referred to as “virtual rates”. It will exploit these virtual rates in the form of the control variables in a new class consisting of rate-based scheduling algorithms. The real end-to end throughput in this algorithm will be represented as Rm. This section presents a joined window-based flow control algorithm and a rate-based scheduling algorithm known as Dynamic Optimized Scheduling and Congestion control in Wireless Sensor Networks [12].

- In this work when a large number of node is placed far away from the BS. Thus, these nodes consume high energy while communicating with the BS and die very quickly.
- Then increases energy consumption during the direct communication of nodes to the BS. And it does not consider fault tolerance issue to avoid the quick failure of overloaded nodes

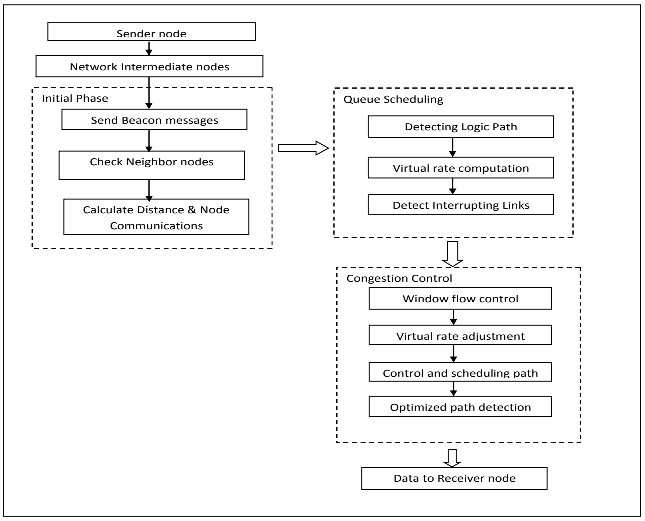

Fig. 2 Methodology chart for proposed system.

2.2 Proposed Methodology

In this work, we used an effective proposed scheme, Dynamic and Optimized Routing Scheme with Secure Transmission in WSN to overcome above many in networks. It provides load balanced and optimized path selection using Secure Fuzzy Logic Optimization and finds a stable path between the source and destination meeting the delay requirement. To conquer security challenge, an active detection-based security and trust routing scheme named ActiveTrust is proposed for WSNs. The most important innovation of ActiveTrust is that it avoids attacks through the active creation of a number of detection routes to quickly detect and obtain nodal trust and thus improve the data route security. Our proposed scheme which maximizes end-to-end connectivity in the network and minimizes faults at link or/and node level. To mitigate this problem in this paper, we proposed a Secure Fuzzy Logic Optimization based routing (SFLOR). FLOR algorithm is designed to maximize the lifetime of WSNs. The algorithm uses a good strategy considering energy levels of the nodes and the lengths of the routed paths. In this paper, we have compared the performance results of our SFLOR approach to the results of the FLOR algorithm. Various differently sized networks are considered, and our approach gives better results than SFLOR algorithm in terms of energy consumption. The main goal of our study was to maintain network life time at a maximum, while discovering the shortest paths from the source nodes to the base node using a fuzzy based optimization technique called SFLOR.

Secure Fuzzy Logic Optimization Routing (SFLOR) where we have taken energy efficiency as major criteria for performing routing and deriving optimized path for data forwarding and processing to base node. The SFLOR generates whole new path of routing by taking energy as fitness value to judge different path and choose best optimized path whose energy consumption is less as compared to other routing paths.

- The out-performs proposed in terms of packet delivery ratio, normalized routing overhead, throughput and average end to end delay.
- Here obtained motion parameters i.e. velocity, direction of the nodes. Based on these parameters the network selects the path to transmit the data packets between the nodes.

- This approach is that best path can be chosen during the routing based on all these factors. Also the battery level of the nodes can be taken care in the network. This results in network's good throughput and high efficiency.
- ActiveTrust can significantly improve the data route success probability and ability against black hole attacks and can optimize network lifetime

2.3 Modules & Descriptions

- Each sensor has 3 parameters (sensorID, groupID, λ) that characterize the sensor. Let λ parameter be the sensor capacity that depends on the sensor's upstream and downstream bandwidth (in Kbps), its number of available links (Available_Con) and its maximum number of links.
- This is because: first, link quality changes significantly when traffic pattern changes and second, link quality estimation takes time to converge, yet different bursts of data traffic are well separated in time, and each burst lasts only for a short period. Beacon-based estimation of link quality is not only limited in reflecting reality, it is also inefficient in energy usage.
- Real time estimation of queues is also needed for ramp control and poses similar problems as intersection queues. The method provably learns the bias, and efficiently estimates the queue length with a theoretical guarantee under a certain condition on the detectors, namely, the stop bar detector reliably indicates when there is no queue in front of it.
- Fuzzy logic is used in this work as main implementation of perceptive reasoning.
- A fuzzy system basically consists of three parts: fuzzifier, fuzzy inference engine, and defuzzifier.
- The fuzzifier maps each crisp input value to the corresponding fuzzy sets and thus assigns it a truth value or degree of membership for each fuzzy set.
- The fuzzifier values are processed by the inference engine, which consists of a rule base and various methods for inferring the rules.
- The objective of our fuzzy-logic-based routing is to determine the energy optimized routing based on the parameters defined previously, such that the network lifetime is maximized.

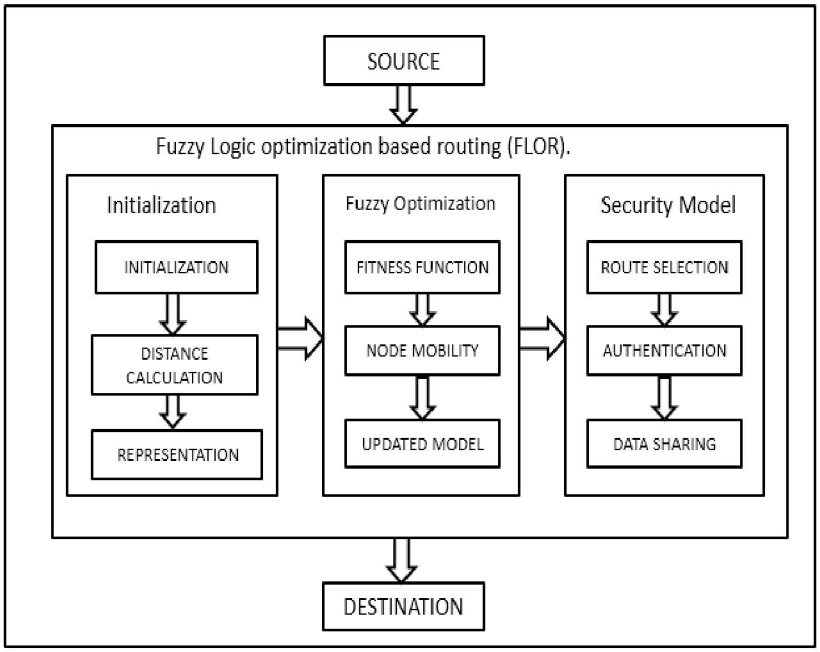

Fig. 3 Flow of fuzzy logic system

Steps involved in the Fuzzy Logic protocol

1. The data are sent by wireless mesh network from source (S) to destination (D) on this network topology.

2. Source node collects the neighbour node list and it transmits the data to destination intermediately through AP (Access Point).
3. APs work together when data sending and receiving process is carried out in the network. The traffic conditions are to be checked at this access point.
4. Fuzzy logic can be applied on this level to the AP and if there is any traffic on this Network path it will select alternate shortest path route to send the data. It mainly works on conditional shortest path routing in the network.
5. It is the more secured method because it is reduces the packet's delay and number of loss packets in this wireless mesh network. The fuzzification works properly at this time of the traffic.
6. f Fuzzy-set logic is applied in some conditions, the data loss can be retrieved from C++ file.
7. If fuzzy operator is executed when the packet loss occurs. Otherwise the Defuzzification process is executed. When data are sent from source to destination, the network finds the shortest path and checks the traffic for data transfer.

Energy Optimized Parameters: We adopt the social welfare function to predict inequality of residual energy of neighbours after selecting different next hop nodes. Based on energy inequality, the method is designed to compute the degree of energy balance. Parameters such as degree of closeness of node to the shortest path, degree of closeness of node to Sink, and degree of energy balance are put into fuzzy logic system. Fuzzy-logic-based energy optimized routing algorithm is proposed to achieve multipara meter, fuzzy routing decision [12].

Energy Consumption Model: The energy consumption of each sensor node consists of three components: sensing energy, communication energy, and data processing energy. Sensing and data processing require much less energy than communication, so we consider only communication energy consumption. If the node transmits an$l$-bit packet over distance$d$, the consumed energy is

$$E_{\mathrm{Tx}}(l,d) = lE_{\mathrm{elec}} + l\varepsilon_{\mathrm{amp}}d^{\alpha}, \qquad (1)$$

When receiving an$l$-bit packet, the energy consumption is

$$E_{\mathrm{Rx}}(l) = lE_{\mathrm{elec}}. \qquad (2)$$

2.4 Degree of Closeness of Node to the Shortest Path.

According to the energy consumption model of sensor nodes, the energy consumption for data transmission is proportional to the square of the distance between the source node and the destination for the free space model. If all relay nodes are on the line from data source node to the Sink, the whole energy consumption for data transmission would be minimized. So, the degree of closeness of node to the shortest path (DCSP) should be used as one of energy optimized parameters. Consider

$$\mathrm{DCSP}(k) = \frac{d(i,\mathrm{Sink})}{d(i,k) + d(k,\mathrm{Sink})}, \qquad (3)$$

Where i denotes source node and$k$denotes its forwarding node, whose distance to Sink is less than i. Note that DCSP (k) attains its maximum (DCSP(k) = 1) when k lies on the line from i to Sink [13].

- Active Trust Detection - A detection route refers to a route without data packets whose goal is to convince the adversary to launch an attack so the system can identify the attack behavior and then mark the black hole location.
- Thus, the system can lower the trust of suspicious nodes and increment the trust of nodes in successful routing routes.
- Through active detection routing, nodal trust can be quickly obtained, and it can effectively guide the data route in choosing nodes with high trust to avoid black holes.
- In this scheme, the source node randomly selects an undetected neighbour node to create an active detection route.

Considering that the longest detection route length is w, the detection route decreases its length by 1 for every hop until the length is decreased to 0, and then the detection route ends. This section details the implementation of the active detection routing protocol.

The content of the detection routing packet can be divided into 6 parts, as shown in Fig. 2: (a)packet head; (b) packet type; (c) ID of the source node; (d) maximum detection route length;(e) acknowledge returned to the source for every w hops; and (f) ID of the packet.Thesourcenodeselectsanundetectednodetolaunchthedetectionroute. Once the detection packet is received by nodes, the maximum route length is decreased by 1. After that, if is 0,generatea feedback packet and launch a feedback route to the source, and then restore to the initial value. If is not 0, then continue to select the next hop in the same way; otherwise, end the route. The structure of a feedback packet is, and it is also composed of6 parts: (a)packet head; (b) packet type; (c) ID of the source node; (d) destination node; (e) ID of the detection packet; and (f) ID of the packet [14].The feedback packet is routed back to the data source; because nodes cache the detection route info, the feedback packet is able to return back to the source, and the following is the algorithm for the detection route protocol. During data routing and detection routing, every node will perform a nodal trust calculation to aid in black hole avoidance. When node A performs a detection route for node B at time it, if the detection data are successfully routed. Node A is the trust evaluator, node C is the target of evaluation, and node B is a recommender of A. Consider B A C to be the direction trust of A to B and C B C to be the direction trust of B to C; then, the recommendation trust of A to C is

$$R_A^C = C_A^B \times C_B^C$$

For the trust of multiple recommendations, the calculation of the recommendation trust from A to B, B to C, etc., until D to E is

$$R_A^E = C_A^B \times C_B^C \times C_C^D \times C_D^E$$

Recommendation trust merging: Consider that the recommender set of node A is AR, in ∈AR and that the recommendation trust of in to node K is, I k A R; then, the merged trust of A to K is

$$U_A^K = \sum_{n_i \in A_n} \left(u_{n_i} R_A^{n_i,k} \right) \; |u_{n_i} = \frac{R_A^{n_i,k}}{\left(R_A^{n_1,k} + R_A^{n_2,k} + \ldots + R_A^{n_{m-1},k} + R_A^{n_m,k} \right)}$$

Comprehensive trust: Comprehensive trust is the total trust, which merges the recommendation trust and direction trust: The comprehensive trust of a node can be computed as follows. After the node launches a detection route, it calculates the direction trust according to Eq. for each received feedback packet. Through interactions, it calculates the direction trust according, and it then calculates the merged trust according to Eq, for the multiple recommendation is the data routing it refers to the process of nodal data routing to the sink. The routing protocol is identical to familiar routing protocols in WSNs; the difference is that the route will select a node with high trust for the next hop. Sensors that are active or asleep are called as surviving sensors and sensors that are malfunctioned. Sensor modes vary, based upon the active sensors vary at each and every time. So, a method to decide a sleep schedule at each and every key time.

III. Implementation Tools.

3.1 Network Simulator (NS)

NS-2 is an open-source simulation tool running on Unix-like operating systems. It is a discreet event simulator targeted at networking research and provides substantial support for simulation of routing, multicast protocols and IP protocols, such as UDP, TCP, RTP and SRM over wired, wireless and satellite networks. It has many advantages that make it a useful tool, such as support for multiple protocols and the capability of graphically detailing network traffic. Additionally, NS-2 supports several algorithms in routing and queuing. LAN routing and broadcasts are part of routing algorithms. Queuing algorithm includes fair queuing, deficit round robin and FIFO. NS-2 started as a variant of the REAL network simulator. REAL is a network simulator originally intended for studying the dynamic behavior of flow and congestion control schemes in packet-switched data networks. In 1995 ns development was supported by Defense Advanced Research Projects Agency DARPA through the VINT project at LBL, Xerox PARC, UCB, and USC/ISI. The wireless code from UCB Daedelus and CMU Monarch projects and Sun Microsystems has added the wireless capabilities to ns-2. NS-2 is available on several platforms such as FreeBSD, Linux, SunOS and Solaris [19]. NS-2 also builds and runs under Windows with Cygwin. Simple scenarios should run on any reasonable machine; however, very large scenarios benefit from large amounts of memory and fast CPU's.

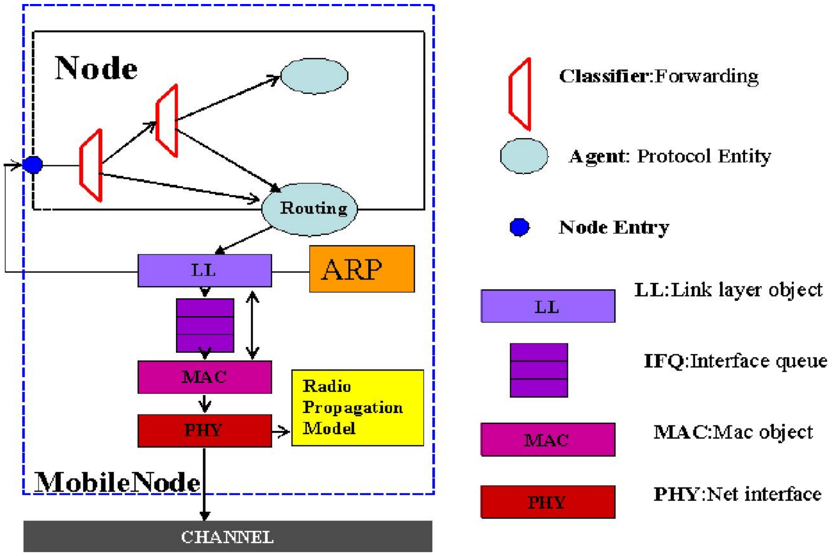

Fig. 4 Schematic Diagram of NS-2

3.2 Architecture of NS-2

As already mentioned above, ns-2 is an object-oriented, discrete event simulator. There are presently five schedulers available in the simulator each of which is implemented by using a different data structure: a simple linked-list, heap, calendar queue (default) and a special type called "real-time". The scheduler runs by selecting the next earliest event, executing it to completion, and returning to execute the next event. The units of time used by the scheduler are seconds. An event is handled by calling the appropriate Handler class.

The most important Handler is NsObject with TclObject as its twin in the OTcl world. They provide all the basic functions allowing objects to interact one with another. For this purpose the receive function group is mainly used. For handling OTcl statements in C++ NsObjects provide the so-called command function. NsObject is the parent class for some important classes as the Classifier, the Connector and the Trace File class. In ns-2 network physical activities are translated to events, events are queued and processed in the order of their scheduled occurrences [20]. And the simulation time progresses with the events processed. And also the simulation "time" may not be the real life time as "inputted". But, why is ns-2 that useful, what kind of work can be done by ns-2, it can model essential network components, traffic models and applications. Typically, it can conFig transport layer protocols, routing protocols, interface queues, and also link layer mechanisms. It can easily see that this software tool in fact could provide us a whole view of the network construction, meanwhile, it also maintain the flexibility for us to decide. Thus, just this one software can help us simulate nearly all parts of the network. This definitely will save us great amount of cost invested on network constructing. The following Fig 3 shows a layered structure which ns-2 can simulate for us.

A. NS Features

- NS is an object-oriented discrete event simulator
  - Simulator maintains list of events and executes one event after another
  - Single thread of control: no locking or race conditions
- Back end is C++ event scheduler
  - Protocols mostly
  - Fast to run, more control
- Front end is TCL
  - Creating scenarios, extensions to C++ protocols
  - fast to write and change

DOI: 10.48175/IJARSCT-5997

IV. Performance Analysis and Conclusion

4.1 End To End Delay

End to end delay is the total time taken to transmit data from sender node to the receiver node involving waiting time, execution time. Fig 5 shows the assessment of preceding and proposed method concern end-to-end delay metric. The number of nodes is taken in x axis and the end-to-end values is taken in y axis. In the previous method, the end-to-end delay values are greater. In presented methodology, the end-to-end delay value is reduced significantly by using the SFLOR method. Consequently, it shows that efficient recognition is performed by using proposed method. Table 1 shows that the comparison statement of end to end delay versus nodes.

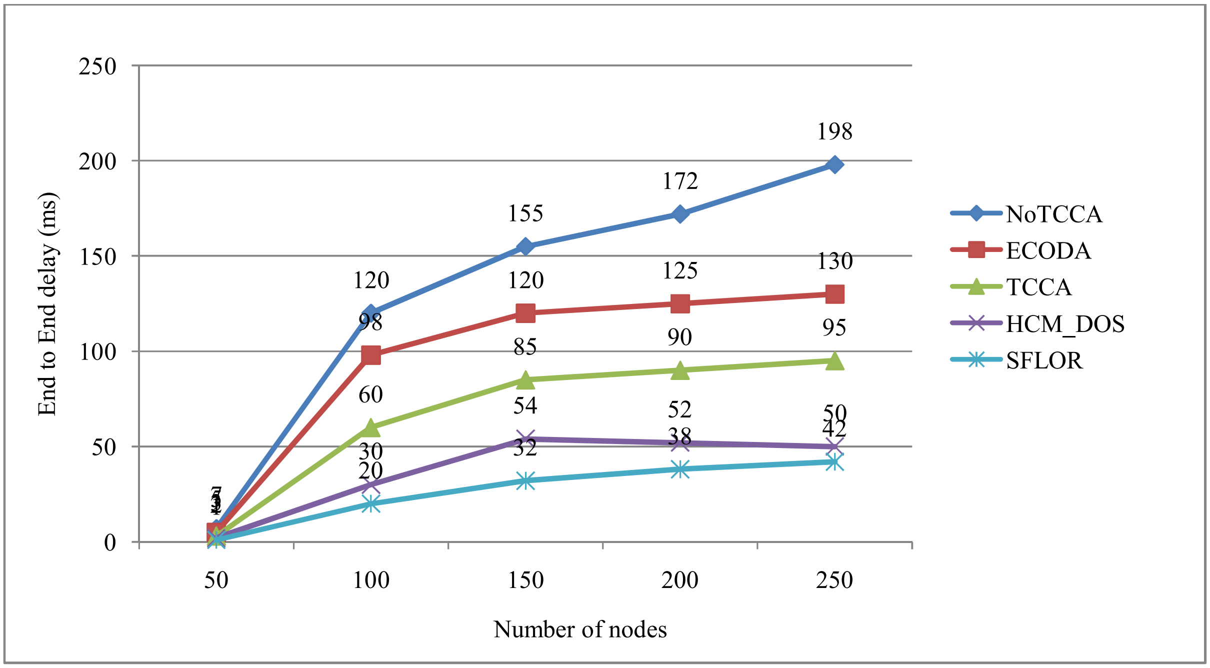

Fig. 5End to End delay evaluations

Table I: End to End Delay (ms) vs Nodes

End to End Delay (ms)					
Nodes	**50**	**100**	**150**	**200**	**250**
NoTCCA	7	120	155	172	198
ECODA	5	98	120	125	130
TCCA	3	60	85	90	95
HCM_DOS	2	30	54	52	50
SFLOR	**1**	**20**	**32**	**38**	**42**

4.2 Network Lifetime

Network Lifetime is called as the time a network performs until the preliminary sensor node or the group of nodes in the network runs away from energy. It can be termed as the entire network lifetime that is discover by the residual energy in the network. Table 2 shows that the comparison statement of Network Lifetime evaluation versus nodes. Consistent with the Fig 6 noticed that the assessment of previous and presented method concern network lifetime metric. The number of nodes is taken in x axis and the network lifetime values is taken in y axis. In previous method, the network lifetime values are lesser

DOI: 10.48175/IJARSCT-5997

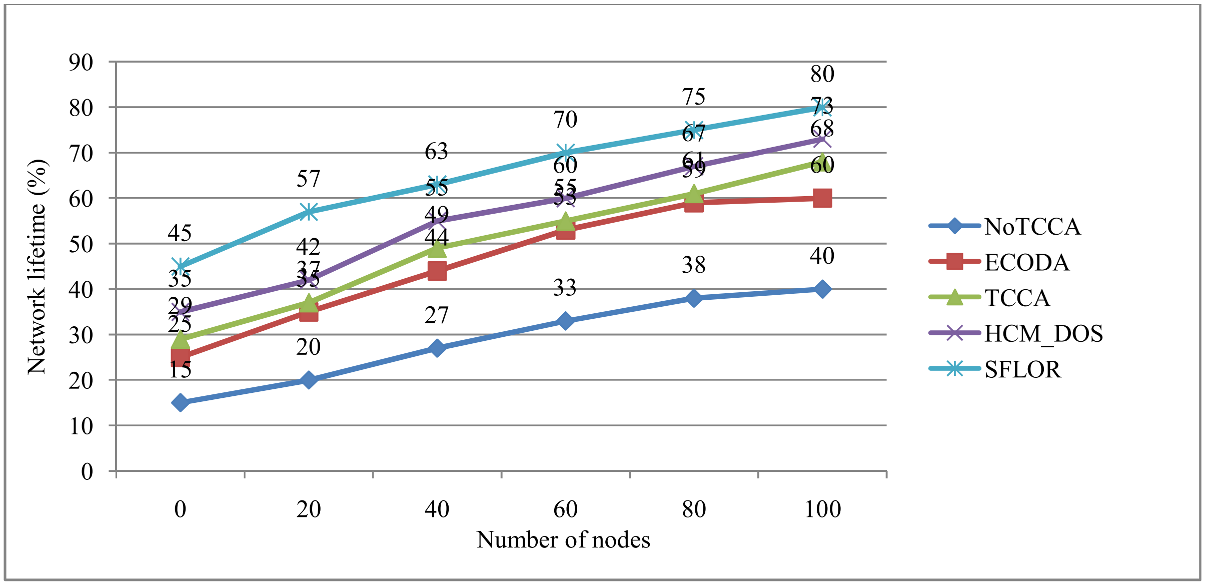

Fig. 6 Network Lifetime evaluations

Table III: Network Lifetime (%) vs Nodes

Network Lifetime (%)						
Nodes	**0**	**20**	**40**	**60**	**80**	**100**
NoTCCA	15	20	27	33	38	40
ECODA	25	35	44	53	59	60
TCCA	29	37	49	55	61	68
HCM_DOS	35	42	55	60	67	73
SFLOR	**45**	**57**	**63**	**70**	**75**	**80**

In presented method, the network lifetime value is improved significantly by using the SFLOR method. Consequently, it shows that efficient identification is performed by using current method.

4.3 Throughput

Network throughput is described as the proportion of successful packet delivery over a message channel.

Table IIIII: Throughput (bits/sec) vs Nodes

Throughput (bits/sec)					
Nodes	**50**	**100**	**150**	**200**	**250**
NoTCCA	0.11	0.21	0.22	0.36	0.4
ECODA	0.17	0.33	0.4	0.51	0.64
TCCA	0.22	0.4	0.55	0.64	0.72
HCM_DOS	0.26	0.53	0.65	0.79	0.86
SFLOR	**0.32**	**0.58**	**0.7**	**0.85**	**0.93**

The throughput is computed in bits per second (bit/s or bps) and the network is known as better when it contains higher throughput. Consistent with the Fig 7 shows that the assessment of previous and presented method concerning throughput metric. The number of nodes is taken in x axis and the throughput values is taken in y axis. In previous method, the throughput values are lesser. Table 3 shows that the comparison statement of Throughput evaluation versus nodes. In presented technique, the throughput value is improved significantly by using the SFLOR method. Consequently, it shows that efficient identification is performed by using proposed method

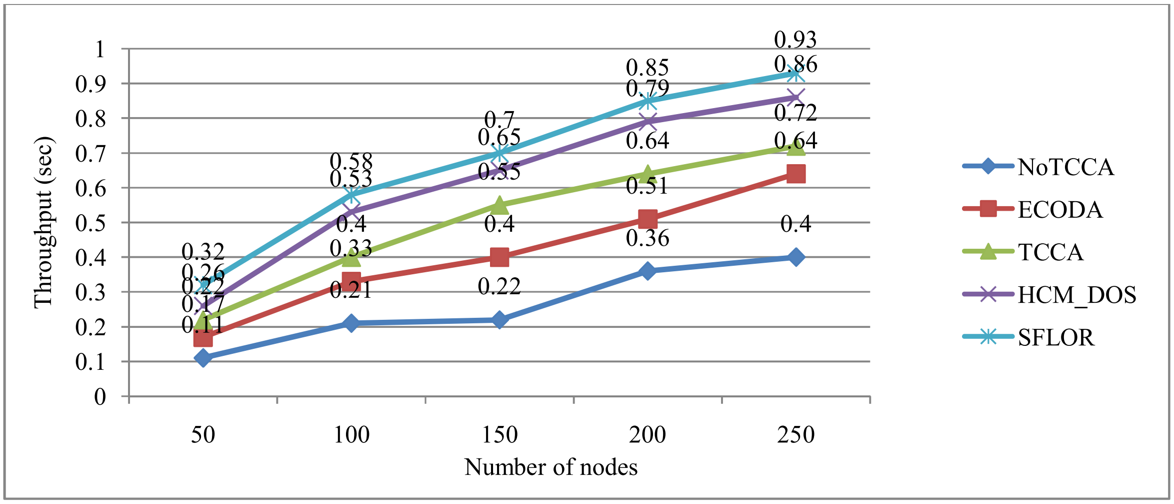

Fig. 7 Throughput evaluations

4.4 Packet Delivery Ratio

Packet delivery ratio is defined as the number of packets productively acquired by the target. From the Fig 8 observed that the assessment of previous and presented method concerning packet delivery ratio. The number of nodes is taken in x axis and the packet delivery ratio values is taken in y axis. In previous method, the packet delivery ratio values are lesser. In presented method, the packet delivery ratio value is improved significantly by using the SFLOR method. Consequently, it shows that efficient identification is performed by using proposed method. Table 4 shows that the comparison statement of packet delivery ratio versus nodes.

Table IVV: Packet Delivery Ratio (%) vs Nodes

	Packet Delivery Ratio (%)				
Nodes	**50**	**100**	**150**	**200**	**250**
NoTCCA	0.12	0.25	0.33	0.46	0.56
ECODA	0.2	0.36	0.43	0.55	0.62
TCCA	0.26	0.42	0.55	0.69	0.75
HCM_DOS	0.29	0.51	0.62	0.75	0.86
SFLOR	**0.35**	**0.58**	**0.67**	**0.83**	**0.92**

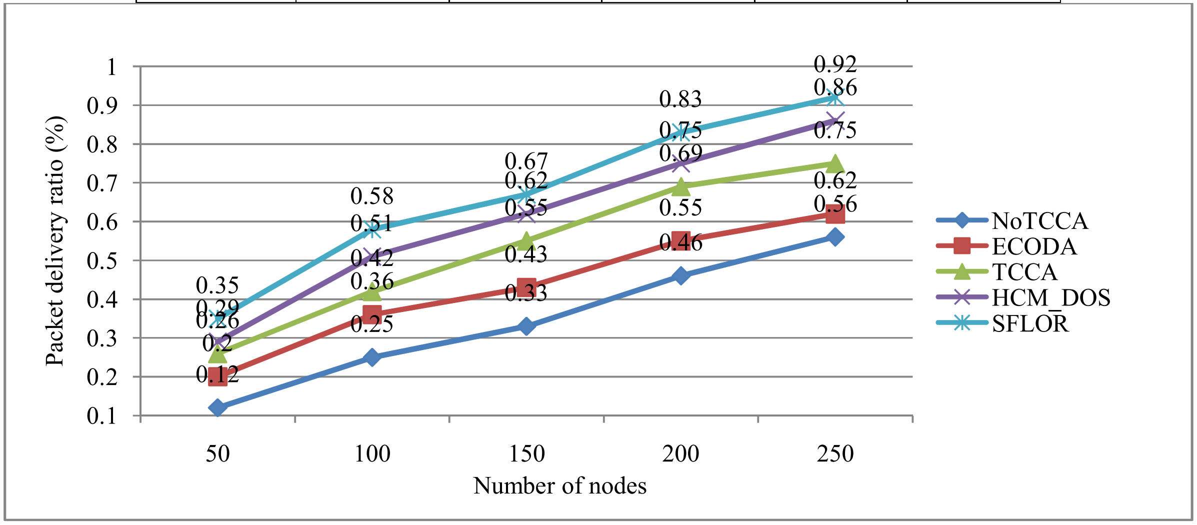

Fig. 8Packet delivery ratio evaluations

4.5 Packet Loss Ratio

The percentage of packets missed for the duration of the transmission is called Packet loss ratio. In Fig 9, whereas the simulation time is 500 seconds, the proposed SFLOR achieves packet loss ratio of 28.59% that is to say 9.59%, 4.95 and 1.04% smaller than DTSN+, and Grid. Table 5 shows that the comparison statement of Packet loss ratio evaluation versus nodes. Therefore, it is superficial that the proposed SFLOR does larger with less packet loss ratio.

Table V: Packet Loss Ratio (%) vs nodes

Packet Loss Ratio (%)					
Nodes	**100**	**200**	**300**	**400**	**500**
NoTCCA	0.13	0.25	0.46	0.7	0.82
ECODA	0.11	0.2	0.35	0.5	0.69
TCCA	0.09	0.1	0.22	0.33	0.45
HCM_DOS	0.05	0.09	0.18	0.24	0.33
SFLOR	**0.03**	**0.05**	**0.11**	**0.18**	**0.25**

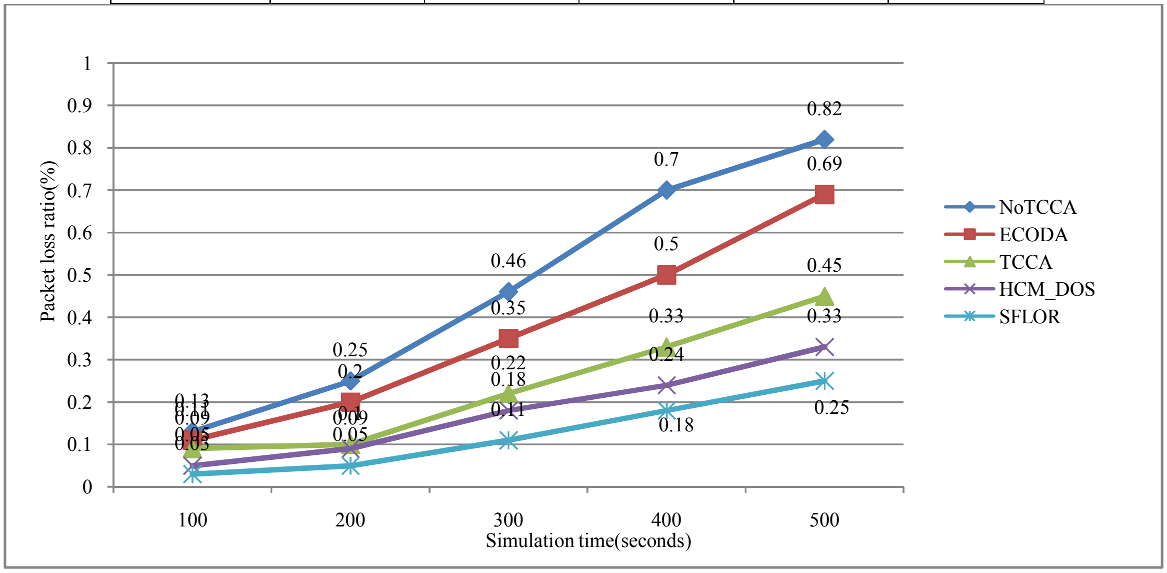

Fig.8 Packet loss ratio evaluation

V. Result

In this paper, we propose Fuzzy Logic Optimization based routing (FLOR). FLOR algorithm is designed to maximize the lifetime of WSNs. The algorithm uses a good strategy considering energy levels of the nodes and the lengths of the routed paths. In this paper, we have compared the performance results of our FLOR approach to the results of the FLOR algorithm. Various differently sized networks are considered, and our approach gives better results than FLOR algorithm in terms of energy consumption. The main goal of our study was to maintain network life time at a maximum, while discovering the shortest paths from the source nodes to the base node using a fuzzy based optimization technique called FLOR.

Fuzzy Logic Optimization Routing (FLOR) where we have taken energy efficiency as major criteria for performing routing and deriving optimized path for data forwarding and processing to base node. The FLOR generates a whole new path of routing by taking energy as fitness value to judge different path and choose best optimized path whose energy consumption is less as compared to other routing paths.

The fuzzy-logic-based routing algorithm is proposed to realize energy optimized, multipara meter, and fuzzy routing decision. Simulation results show that the algorithm extends the network lifetime effectively compared with similar algorithms for different dada generation patterns and has a good performance in terms of energy balance and energy efficiency.

REFERENCES

[1]. Abbas, Nasim, and Fengqi Yu. "A traffic congestion control algorithm for wireless multimedia sensor networks", In 2018 IEEE SENSORS, pp. 1-4. IEEE, 2018.

[2]. Heinzelman, Wendi Rabiner, AnanthaChandrakasan, and Hari Balakrishnan. "Energy-efficient communication protocol for wireless microsensor networks." In Proceedings of the 33rd annual Hawaii international conference on system sciences, pp. 10-pp. IEEE, 2000.

[3]. Gupta, Gaurav, and Mohamed Younis. "Fault-tolerant clustering of wireless sensor networks." In 2003 IEEE Wireless Communications and Networking, 2003. WCNC 2003., vol. 3, pp. 1579-1584. IEEE, 2003.

[4]. Jiang, Chang-Jiang, Wei-Ren Shi, and Xian-lun TANG. "Energy-balanced unequal clustering protocol for wireless sensor networks." The Journal of China Universities of Posts and Telecommunications 17, no. 4 (2010): 94-99.

[5]. Latiff, NM Abdul, Charalampos C. Tsimenidis, and Bayan S. Sharif. "Energy-aware clustering for wireless sensor networks using particle swarm optimization." In 2007 IEEE 18th international symposium on personal, indoor and mobile radio communications, pp. 1-5. IEEE, 2007.

[6]. Rao, P. C., Prasanta K. Jana, and Haider Banka. "A particle swarm optimization based energy efficient cluster head selection algorithm for wireless sensor networks." Wireless networks 23, no. 7 (2017): 2005-2020.

[7]. Alipio, Melchizedek, Nestor Michael Tiglao, Antonio Grilo, Fawaz Bokhari, Umair Chaudhry, and Shavez Qureshi. "Cache-based transport protocols in wireless sensor networks: A survey and future directions." Journal of Network and Computer Applications 88 (2017): 29-49.

[8]. Aslam, Nelofar, Kewen Xia, Ahmad Ali, and Saleem Ullah. "Adaptive TCP-ICCW congestion control mechanism for QoS in renewable wireless sensor networks." IEEE sensors letters 1, no. 6 (2017): 1-4.

[9]. Alipio, Melchizedek I., and Nestor Michael C. Tiglao. "Analysis of cache-based transport protocol at congestion in wireless sensor networks." In 2017 International Conference on Information Networking (ICOIN), pp. 360-365. IEEE, 2017.

[10]. Tiglao, Nestor Michael C., and António M. Grilo. "Cross-layer caching based optimization for wireless multimedia sensor networks." In 2012 IEEE 8th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), pp. 697-704. IEEE, 2012.

[11]. Tiglao, Nestor Michael C., and António M. Grilo. "An analytical model for transport layer caching in wireless sensor networks." Performance Evaluation 69, no. 5 (2012): 227-245.

[12]. Alipio, Melchizedek I., and Nestor Michael C. Tiglao. "A cache-aware congestion control for reliable transport in wireless sensor networks." In International Conference on Mobile Networks and Management, pp. 217-230. Springer, Cham, 2017.

[13]. Kafi, Mohamed Amine, Djamel Djenouri, Jalel Ben-Othman, and NadjibBadache. "Congestion control protocols in wireless sensor networks: a survey." IEEE communications surveys & tutorials 16, no. 3 (2014): 1369-1390.

[14]. Tiglao, Nestor Michael C., and António M. Grilo. "Transmission window optimization for caching-based transport protocols in wireless sensor networks." In International Wireless Internet Conference, pp. 39-46. Springer, Cham, 2014.

[15]. Grilo, António M., and Mike Heidrich. "Routing metrics for cache-based reliable transport in wireless sensor networks." EURASIP Journal on Wireless Communications and Networking 2013, no. 1 (2013): 1-16.

[16]. Meneses, Duarte, António Grilo, and Paulo Rogério Pereira. "A transport protocol for real-time streaming in wireless multimedia sensor networks." In 2011 7th EURO-NGI Conference on Next Generation Internet Networks, pp. 1-8. IEEE, 2011.

[17]. N. M. C. Tiglao and A. M. Grilo, "Cross-layer caching based optimization for wireless multimedia sensor networks," in Wireless and Mobile Computing, Networking and Communications (WiMob), 2012 IEEE 8th International Conference on,Oct2012,pp.697–704.

[18]. Tiglao, Nestor Michael C., and António M. Grilo. "An analytical model for transport layer caching in wireless sensor networks." Performance Evaluation 69, no. 5 (2012): 227-245.

DOI: 10.48175/IJARSCT-5997

[19]. Alipio, Melchizedek I., and Nestor Michael C. Tiglao. "A cache-aware congestion control for reliable transport in wireless sensor networks." In International Conference on Mobile Networks and Management, pp. 217-230. Springer, Cham, 2017.

[20]. Wei, David X., and Pei Cao. "NS-2 TCP-Linux: an NS-2 TCP implementation with congestion control algorithms from Linux." In Proceeding from the 2006 workshop on ns-2: the IP network simulator, pp. 9-es. 2006.

[21]. Patel, Sanjeev, P. K. Gupta, Arjun Garg, Prateek Mehrotra, and Manish Chhabra. "Comparative analysis of congestion control algorithms using ns-2." arXiv preprint arXiv:1203.3654 (2012).

[22]. Kim, Tae-Woon, Sang-Hwa Chung, In-Su Yoon, and Jeong-Soo Kim. "Effects of handover on TCP congestion control algorithms over mobile WiMAX." In 2008 5th IEEE Consumer Communications and Networking Conference, pp. 1230-1231. IEEE, 2008.

DOI: 10.48175/IJARSCT-5997